

# Simulated Umbrella Trial

Akif Mustafa

2024-09-25

## Baseline data

```
#####  
  
# Basket Trial with Bayesian Adaptive Design #  
  
library(tidyverse)  
  
## Warning: package 'tidyverse' was built under R version 4.3.3  
  
## Warning: package 'tidyr' was built under R version 4.3.3  
  
## Warning: package 'purrr' was built under R version 4.3.3  
  
## Warning: package 'lubridate' was built under R version 4.3.3  
  
## — Attaching core tidyverse packages — tidyverse 2.0.0 —  
## ✓ dplyr      1.1.4      ✓ readr      2.1.4  
## ✓ forcats   1.0.0      ✓ stringr    1.5.1  
## ✓ ggplot2   3.4.4      ✓ tibble     3.2.1  
## ✓ lubridate 1.9.3      ✓ tidyr      1.3.1  
## ✓ purrr     1.0.2  
## — Conflicts — tidyverse_conflicts() —  
## ✗ dplyr::filter() masks stats::filter()  
## ✗ dplyr::lag()     masks stats::lag()  
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be  
come errors
```

```
# Set number of individuals
n <- 540

# Create a data frame with ids and random treatment assignment
set.seed(123)

baseline <- data.frame(
  id = 1:540,
  treatment = sample(c("A", "B", "C", "D", "S"), n, replace = TRUE),
  fev_fvc_ratio = pmin(0.95, pmax(0.20, rnorm(n, mean = 0.75, sd = 0.15))), # Simulating FEV
  values
  SGRQ = rnorm(n, mean = 40, sd = 15), # Simulating SGRQ score
  age = sample(40:85, n, replace = TRUE), # Random ages between 40 and 85
  sex = sample(c("M", "F"), n, replace = TRUE), # Random gender
  BMI = rnorm(n, mean = 26, sd = 5) # Simulating BMI
)

#summary of the baseline data
base_summary <- baseline %>%
  group_by(treatment) %>%
  summarise(
    N = n(),
    Males = sum(sex == "M"),
    Females = sum(sex == "F"),
    mean_BMI = mean(BMI, na.rm = T),
    SD_BMI = sd(BMI, na.rm = T),
    mean_ratio = mean(fev_fvc_ratio, na.rm = T),
    SD_ratio = sd(fev_fvc_ratio, na.rm = T),
    mean_SGRQ = mean(SGRQ, na.rm = T)
  )

print(base_summary)
```

```
## # A tibble: 5 × 9
##   treatment      N Males Females mean_BMI SD_BMI mean_ratio SD_ratio mean_SGRQ
##   <chr>      <int> <int>   <int>   <dbl> <dbl>      <dbl>    <dbl>    <dbl>
## 1 A          115    48     67    25.7  4.94      0.740    0.148    43.6
## 2 B          110    57     53    26.3  4.22      0.737    0.142    39.3
## 3 C          104    49     55    25.7  4.95      0.737    0.136    40.8
## 4 D          102    50     52    25.4  5.07      0.766    0.130    37.5
## 5 S          109    60     49    25.7  4.87      0.749    0.136    40.9
```

## First Interim Analysis

```
library(tidyverse)
library(MCMCpack)
```

```
## Warning: package 'MCMCpack' was built under R version 4.3.3
```

```
## Loading required package: coda
```

```
## Loading required package: MASS
```

```
##  
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':  
##  
##      select
```

```
## Warning in .recacheSubclasses(def@className, def, env): undefined subclass  
## "ndiMatrix" of class "replValueSp"; definition not updated
```

```
## ##  
## ## Markov Chain Monte Carlo Package (MCMCpack)
```

```
## ## Copyright (C) 2003-2024 Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park
```

```
## ##  
## ## Support provided by the U.S. National Science Foundation
```

```
## ## (Grants SES-0350646 and SES-0350613)  
## ##
```

```
# First follow-up data before 1st interim analysis

#deaths
# A = 9
# B = 8
# C = 6
# D = 7
# S = 8

#Comparing drug A vs SC

d.A = 10
n.A = 115
d.S = 8
n.S = 109

#prior information (weekly informative)
alpha.A = 1
beta.A = 1
alpha.S = 1
beta.S = 1

#posterior distribution parameters
post.alpha.A = alpha.A + d.A
post.beta.A = beta.A + n.A - d.A

post.alpha.S = alpha.S + d.S
post.beta.S = beta.S + n.S - d.S

#generate posterior samples
set.seed(555)
post.dist.A = rbeta(10000, post.alpha.A, post.beta.A)
post.dist.S = rbeta(10000, post.alpha.S, post.beta.S)

# Estimate the probability that Drug A is better (lower mortality) than SC

p.A.better = mean(post.dist.A < post.dist.S)
p.A.better
```

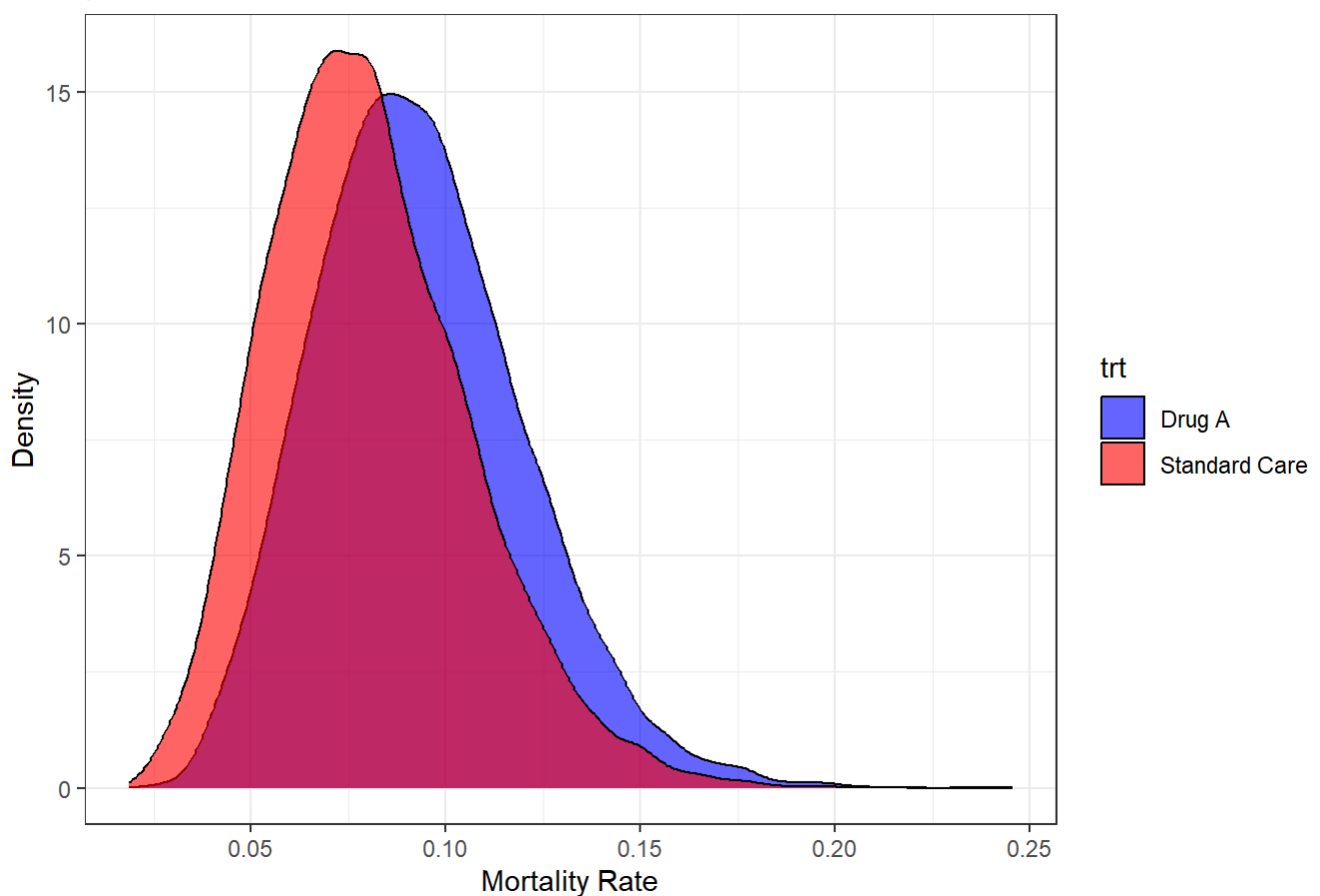
```
## [1] 0.3648
```

*#Plotting the data*

```
data.A.S = data.frame(
  d.r = c(post.dist.A , post.dist.S),
  trt = rep(c("A", "S"), times = c(length(post.dist.A), length(post.dist.S)))
)

ggplot(data.A.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment A ans SC",
       x = "Mortality Rate",
       y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug A", "Standard Care"))+
  theme_bw()
```

posterior distribution of death rate in treatment A ans SC

*#using MCMCpack*

```
set.seed(555)
post.A = MCbinomialbeta(d.A, n.A, alpha.A, beta.A, mcmc = 10000)
post.S = MCbinomialbeta(d.S, n.S, alpha.S, beta.S, mcmc = 10000)

mean(post.A < post.S)
```

```
## [1] 0.348
```

```
#####
```

```
#Comparing drug B vs SC
```

```
d.B = 8
n.B = 110
```

```
#prior
alpha.B = 1
beta.B = 1
```

```
#posterior
post.alpha.B = alpha.B + d.B
post.beta.B = beta.B + n.B - d.B
```

```
#generating posterior samples
set.seed(555)
post.dist.B = rbeta(10000, post.alpha.B, post.beta.B)
```

```
#probability that drug B is better than SC
```

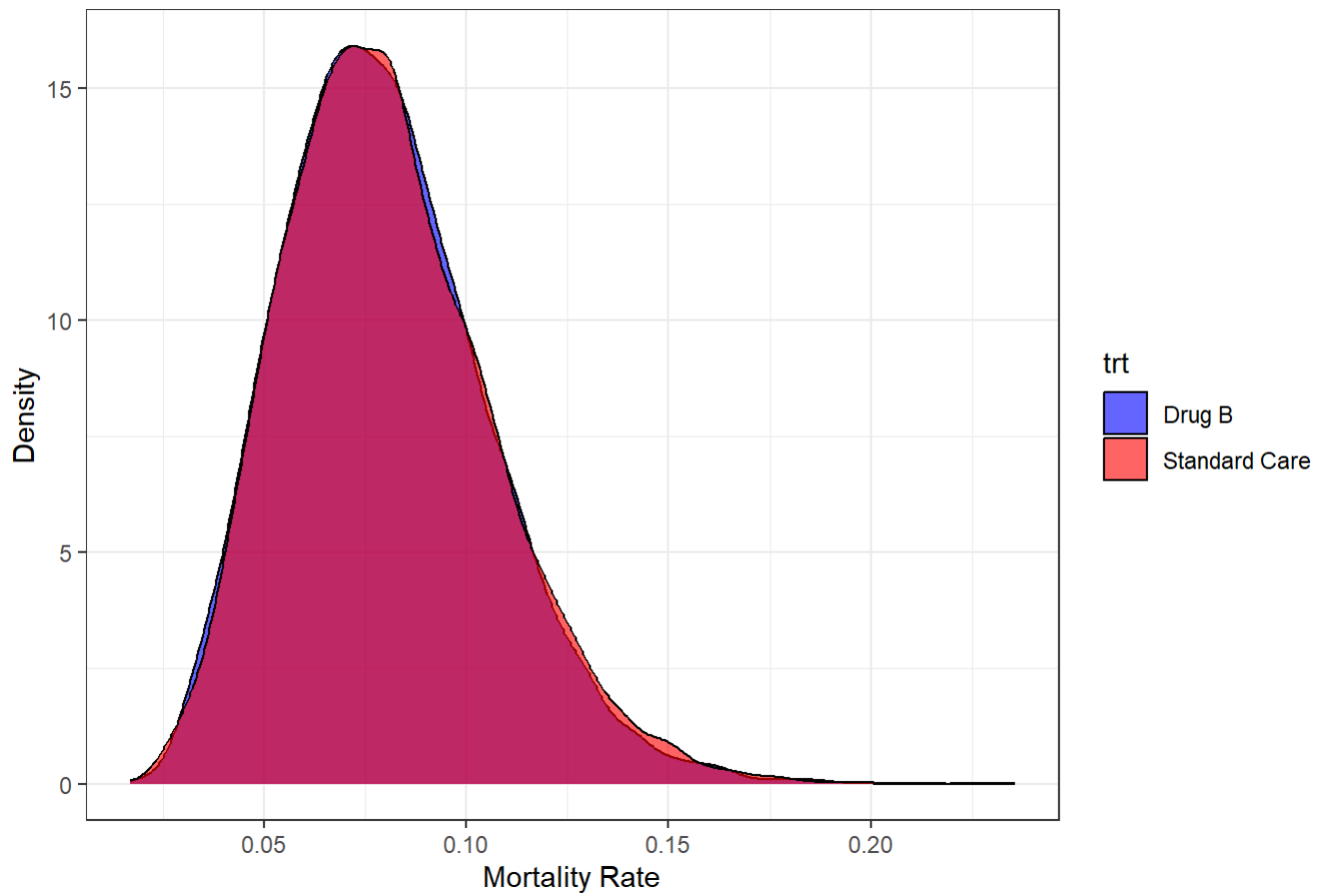
```
p.B.better = mean(post.dist.B < post.dist.S)
print(p.B.better)
```

```
## [1] 0.5019
```

```
#plotting distributions
data.B.S = data.frame(
  d.r = c(post.dist.B , post.dist.S),
  trt = rep(c("B", "S"), times = c(length(post.dist.B), length(post.dist.S)))
)

ggplot(data.B.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment B ans SC",
    x = "Mortality Rate",
    y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug B", "Standard Care"))+
  theme_bw()
```

## posterior distribution of death rate in treatment B ans SC



```
#using MCMCpack
set.seed(555)
post.B = MCbinomialbeta(d.B , n.B, post.alpha.B, post.beta.B, mcmc = 10000)
mean(post.B < post.S)
```

```
## [1] 0.548
```

```
#####
```

```
#Comparing drug C vs SC
```

```
d.C = 6
```

```
n.C = 104
```

```
#prior
```

```
alpha.C = 1
```

```
beta.C = 1
```

```
#posterior
```

```
post.alpha.C = alpha.C + d.C
```

```
post.beta.C = beta.C + n.C - d.C
```

```
#generating samples
```

```
set.seed(555)
```

```
post.dist.C = rbeta(10000, post.alpha.C, post.beta.C)
```

```
p.C.better = mean(post.dist.C < post.dist.S)
```

```
print(p.C.better)
```

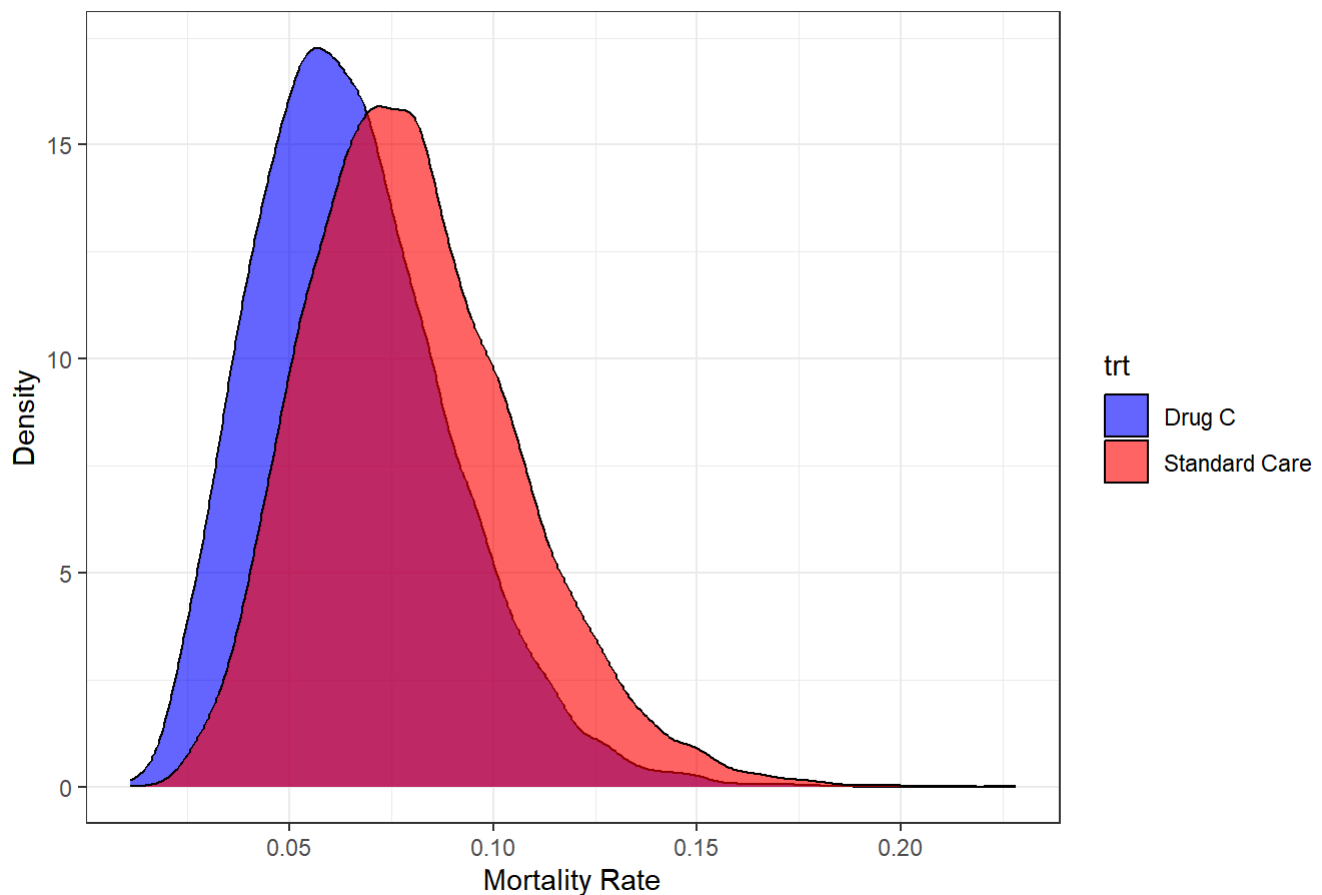
```
## [1] 0.6643
```

```
data.C.S = data.frame(  
  d.r = c(post.dist.C , post.dist.S),  
  trt = rep(c("C", "S"), times = c(length(post.dist.C), length(post.dist.S)))  
)
```

```
ggplot(data.C.S, aes(x = d.r , fill = trt)) +  
  geom_density(alpha = 0.6) +  
  labs(title = "posterior distribution of death rate in treatment C ans SC",  
        x = "Mortality Rate",  
        y = "Density")+  
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug C", "Standard Care"))+  
  theme_bw()
```



## posterior distribution of death rate in treatment C ans SC



```
#using MCMCpack
set.seed(555)
post.C = MCbinomialbeta(d.C, n.C, post.alpha.C, post.beta.C, mcmc = 10000)
mean(post.C < post.S)
```

```
## [1] 0.741
```

```
#####
#comparing drug D vs SC

d.D = 7
n.D = 102

alpha.D = 1
beta.D = 1

post.alpha.D = alpha.D + d.D
post.beta.D = beta.D + n.D - d.D

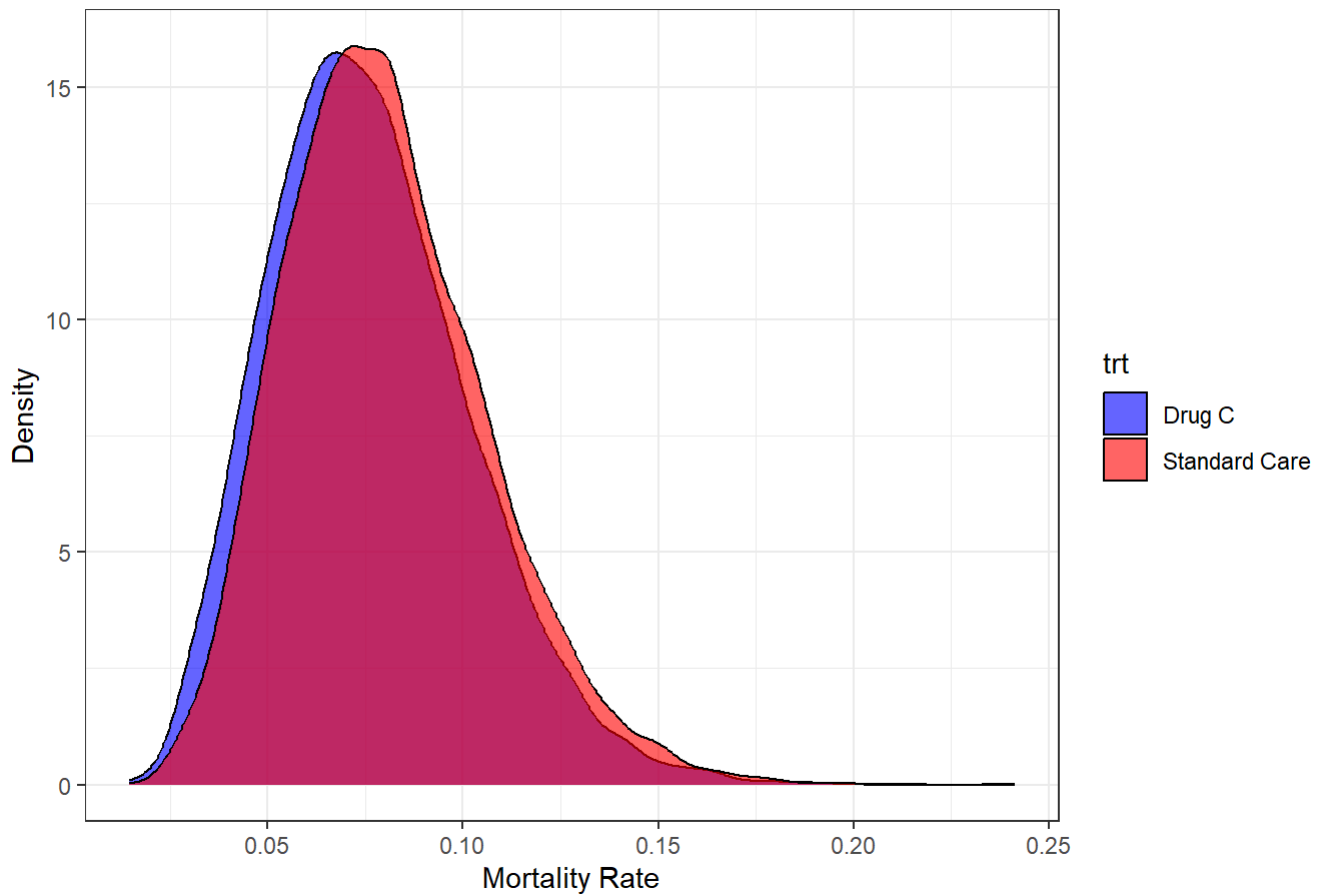
set.seed(555)
post.dist.D = rbeta(10000, post.alpha.D, post.beta.D)
p.D.better = mean(post.dist.D < post.dist.S)
print(p.D.better)
```

```
## [1] 0.5458
```

```
data.D.S = data.frame(
  d.r = c(post.dist.D , post.dist.S),
  trt = rep(c("D", "S"), times = c(length(post.dist.D), length(post.dist.S)))
)

ggplot(data.D.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment C ans SC",
       x = "Mortality Rate",
       y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug C", "Standard Care"))+
  theme_bw()
```

posterior distribution of death rate in treatment C ans SC



```
#using MCMCpack
#using MCMCpack
set.seed(555)
post.D = MCbinomialbeta(d.D, n.D, post.alpha.D, post.beta.D, mcmc = 10000)
mean(post.D < post.S)
```

```
## [1] 0.598
```

```
#####
```

```
#updating randomization probabilities for second round of recruitment
#fixing probability of randomization in SC to 0.20
```

```
p.better = c(p.A.better, p.B.better, p.C.better, p.D.better)
p.rand1 = (p.better/sum(p.better))*0.8
p.rand1 = c((p.rand1), S = 0.20)
print(p.rand1)
```

```
##                               S
## 0.1405239 0.1933359 0.2558937 0.2102465 0.2000000
```

## Second Interim Analysis

```
library(tidyverse)
library(MCMCpack)
```

```
#####
```

```
#second interim analysis#
```

```
#Futility criteria: if the probability that drug is better than SC is less than 0.40 then the
drug will be considered futile
```

```
#Efficacy criteria: If the probability that drug is better than SC is more than 0.95 then the
drugs will be concluded better than SC
```

```
#treatment allocation in second round of recruitment (810 patients)
```

```
treatment2 = sample(c("A", "B", "C", "D", "S"), 810, prob = c(p.rand1), replace = TRUE)
table(treatment2)
```

```
## treatment2
##   A    B    C    D    S
##  98 171 217 161 163
```

```

#total patients in each arm (recruited)
#A = 115 + 98 = 213
#B = 110 + 171 = 281
#c = 104 + 217 = 321
#D = 102 + 161 = 263
#S = 109 + 163 = 272

#total deaths in each arm at seconf follow-up (before second interim analysis)
# A = 32      (15%)
# B = 34      (12.8)
# C = 27      (8.4%)
# D = 29      (11%)
# S = 31      (11.3%)

#####

#drug A vs SC

d2.A = 32
n2.A = 213

d2.S = 31
n2.S = 272

#prior information from posterior of previous intermin analysis

alpha2.A = post.alpha.A
beta2.A = post.beta.A

alpha2.S = post.alpha.S
beta2.S = post.beta.S

#updated posterior parameters

post2.alpha.A = alpha2.A + d2.A
post2.beta.A = beta2.A + n2.A - d2.A

post2.alpha.S = alpha2.S + d2.S
post2.beta.S = beta2.S + n2.S - d2.S

#Generating samples
set.seed(555)
n = 10000
postdist2.A = rbeta(n, post2.alpha.A, post2.beta.A)
postdist2.S = rbeta(n, post2.alpha.S, post2.beta.S)
p2.A.better = mean(postdist2.A < postdist2.S)
print(p2.A.better)

```

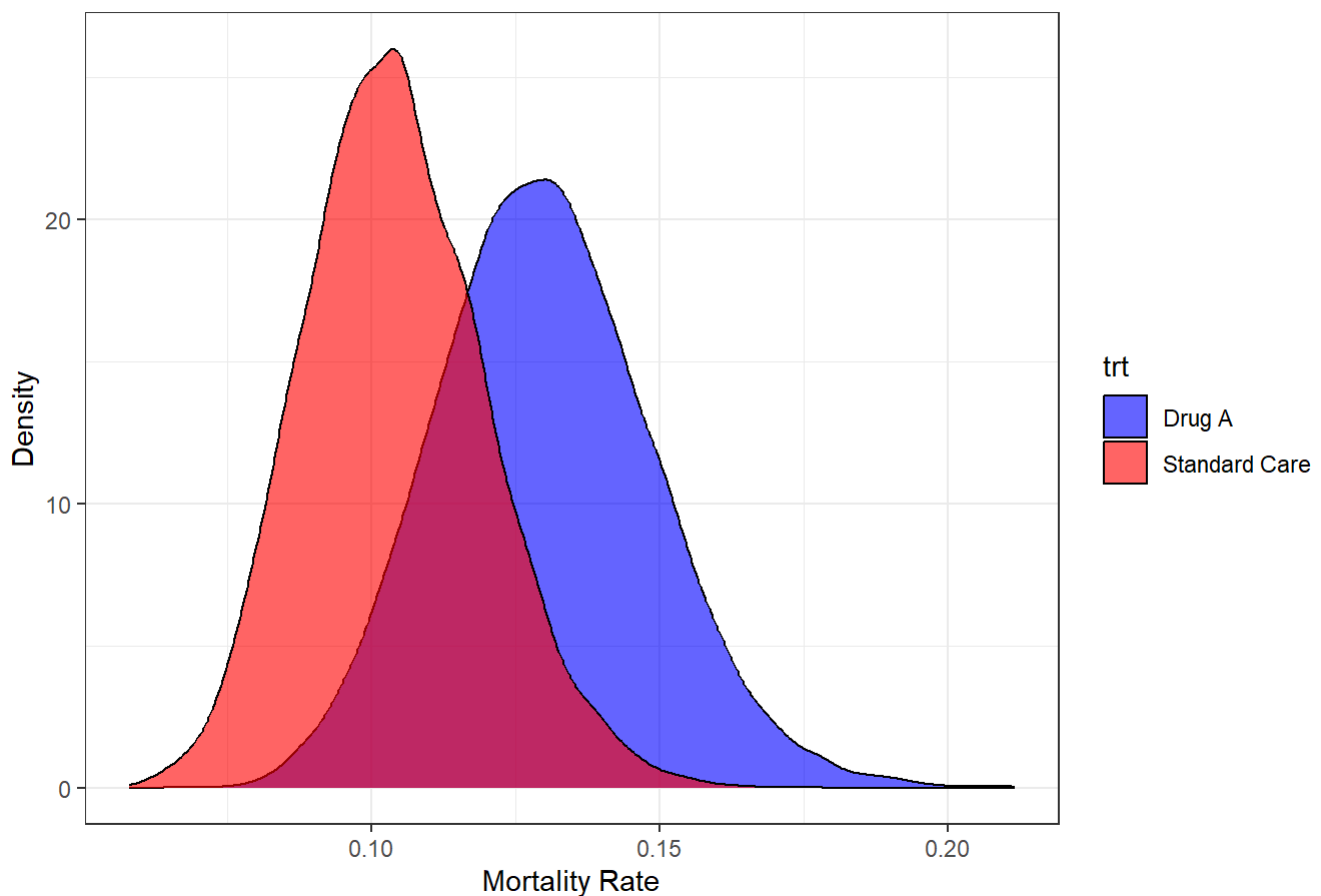
```
## [1] 0.1443
```

*#the probability is less than 0.40. hence treatment A is concluded to be futile*

```
data2.A.S = data.frame(
  d.r = c(postdist2.A , postdist2.S),
  trt = rep(c("A", "S"), times = c(n, n))
)

ggplot(data2.A.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment A ans SC",
    x = "Mortality Rate",
    y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug A", "Standard Care"))+
  theme_bw()
```

posterior distribution of death rate in treatment A ans SC



*#using MCMCpack*

```
post2.A = MCBinomialbeta(d2.A, n2.A, post2.alpha.A, post2.beta.A, mcmc = n)
post2.S = MCBinomialbeta(d2.S, n2.S, post2.alpha.S, post2.beta.S, mcmc = n)
mean(post2.A < post2.S)
```

```
## [1] 0.051
```

```
#####
```

```
#drug B vs SC
```

```
d2.B = 34
```

```
n2.B = 281
```

```
alpha2.B = post.alpha.B
```

```
beta2.B = post.beta.B
```

```
post2.alpha.B = alpha2.B + d2.B
```

```
post2.beta.B = beta2.B + n2.B - d2.B
```

```
postdist2.B = rbeta(n , post2.alpha.B, post2.beta.B)
```

```
p2.B.better = mean(postdist2.B < postdist2.S)
```

```
print(p2.B.better)
```

```
## [1] 0.4041
```

```
data2.B.S = data.frame(
```

```
  d.r = c(postdist2.B , postdist2.S),
```

```
  trt = rep(c("B", "S"), times = c(n, n))
```

```
)
```

```
ggplot(data2.B.S, aes(x = d.r , fill = trt)) +
```

```
  geom_density(alpha = 0.6) +
```

```
  labs(title = "posterior distribution of death rate in treatment B ans SC",
```

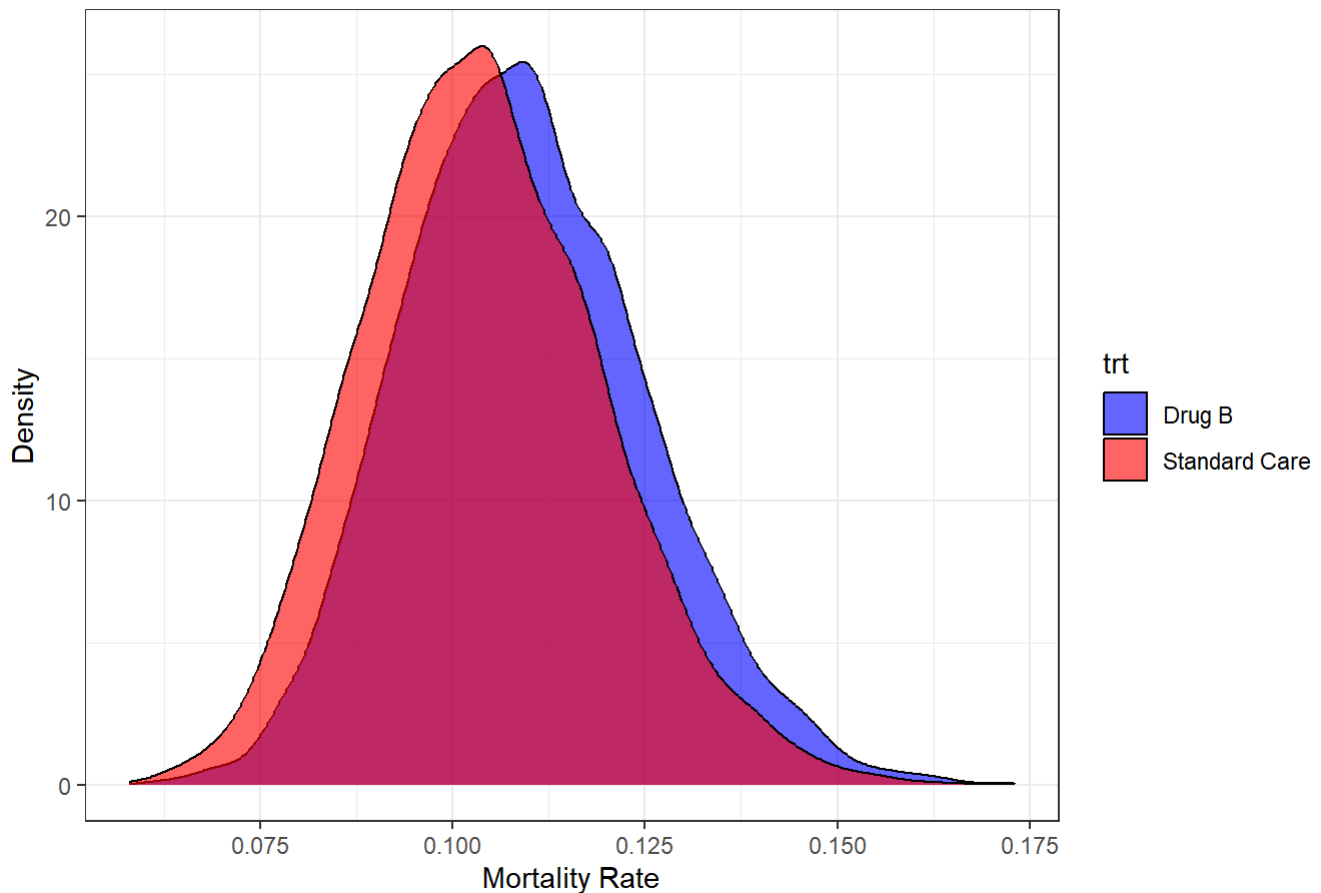
```
        x = "Mortality Rate",
```

```
        y = "Density")+
```

```
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug B", "Standard Care"))+
```

```
  theme_bw()
```

## posterior distribution of death rate in treatment B ans SC



```
#using MCMCpack
```

```
post2.B = MCbinomialbeta(d2.B, n2.B, post2.alpha.B, post2.beta.B, mcmc = n)
```

```
mean(post2.B < post2.S)
```

```
## [1] 0.368
```

```
#####
```

```
#drug C vs SC
```

```
d2.C = 27
```

```
n2.C = 321
```

```
alpha2.C = post.alpha.C
```

```
beta2.C = post.beta.C
```

```
post2.alpha.C = alpha2.C + d2.C
```

```
post2.beta.C = beta2.C + n2.C - d2.C
```

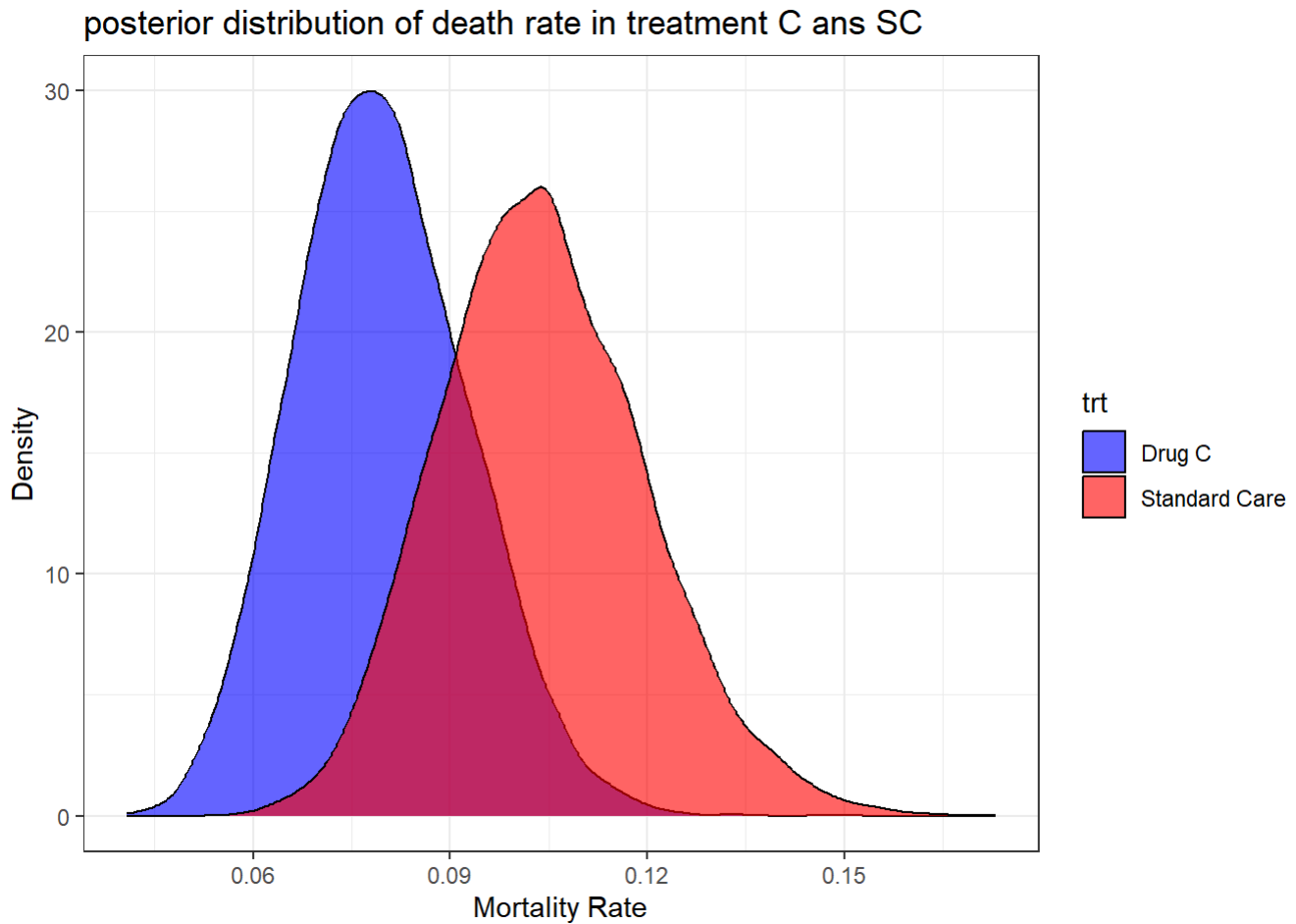
```
postdist2.C = rbeta(n , post2.alpha.C, post2.beta.C)
```

```
p2.C.better = mean(postdist2.C < postdist2.S)
```

```
print(p2.C.better)
```

```
## [1] 0.8863
```

```
data2.C.S = data.frame(  
  d.r = c(postdist2.C , postdist2.S),  
  trt = rep(c("C", "S"), times = c(n, n))  
)  
  
ggplot(data2.C.S, aes(x = d.r , fill = trt)) +  
  geom_density(alpha = 0.6) +  
  labs(title = "posterior distribution of death rate in treatment C ans SC",  
    x = "Mortality Rate",  
    y = "Density")+  
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug C", "Standard Care"))+  
  theme_bw()
```



```
#using MCMCpack  
post2.C = MCbinomialbeta(d2.C, n2.C, post2.alpha.C, post2.beta.C, mcmc = n)  
mean(post2.C < post2.S)
```

```
## [1] 0.96
```



```
#####
```

```
#drug D vs SC 31 263
```

```
d2.D = 31
n2.D = 263

alpha2.D = post.alpha.D
beta2.D = post.beta.D

post2.alpha.D = alpha2.D + d2.D
post2.beta.D = beta2.D + n2.D - d2.D

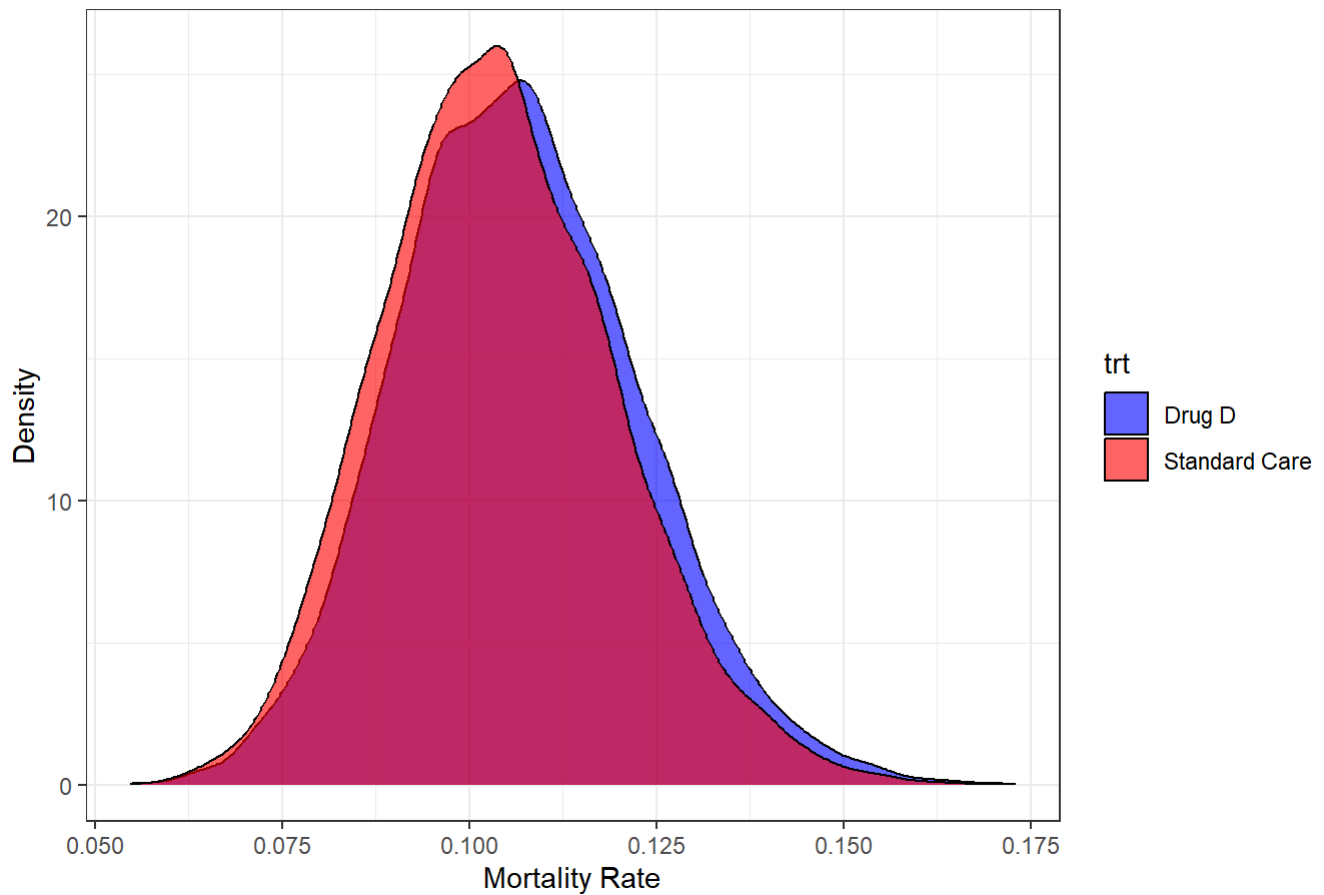
postdist2.D = rbeta(n , post2.alpha.D, post2.beta.D)
p2.D.better = mean(postdist2.D < postdist2.S)
print(p2.D.better)
```

```
## [1] 0.4565
```

```
data2.D.S = data.frame(
  d.r = c(postdist2.D , postdist2.S),
  trt = rep(c("D", "S"), times = c(n, n))
)

ggplot(data2.D.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment D ans SC",
    x = "Mortality Rate",
    y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug D", "Standard Care"))+
  theme_bw()
```

## posterior distribution of death rate in treatment D and SC



```
#using MCMCpack
```

```
post2.D = MCBinomialbeta(d2.D, n2.D, post2.alpha.D, post2.beta.D, mcmc = n)
```

```
mean(post2.D < post2.S)
```

```
## [1] 0.42
```

```
#updated randomization probability
```

```
# Arm A has been dropped
```

```
p2.better = c(p2.B.better, p2.C.better, p2.D.better)
```

```
p.rand2 = (p2.better/sum(p2.better))*0.75
```

```
p.rand2 = c((p.rand2), S = 0.25)
```

```
print(p.rand2)
```

```
##                               S
## 0.1734930 0.3805169 0.1959900 0.2500000
```

## Third Interim Analysis

```
library(tidyverse)
library(MCMCpack)
```

```
#####
```

```
# 3rd Interim Analysis #
```

```
# Futility criteris: Futile if  $P(\text{better}) < 0.70$ 
```

```
# Efficacy criteria: Better than SC is  $P(\text{better}) > 0.975$ 
```

```
# Equivalence: if  $0.80 < P(\text{better}) \leq 0.975$ 
```

```
#treatment allocation in third round of recruitment (860 patients)
```

```
treatment3 = sample(c("B", "C", "D", "S"), 860, prob = c(p.rand2), replace = TRUE)
```

```
table(treatment3)
```

```
## treatment3
```

```
##   B    C    D    S
```

```
## 154 304 175 227
```

```
#total patients in each arm (recruited)
```

```
#B = 281 + 147 = 428
```

```
#C = 321 + 328 = 649
```

```
#D = 263 + 179 = 442
```

```
#S = 272 + 206 = 478
```

```
#total deaths in each arm at third follow-up (before second interim analysis)
```

```
# B = 60      (14%)
```

```
# C = 59      (9%)
```

```
# D = 71      (16%)
```

```
# S = 66      (13.8%)
```

```
#####
```

```
#drug B vs SC
```

```
d3.B = 60
```

```
n3.B = 428
```

```
d3.S = 66
```

```
n3.S = 478
```

```
#prior
```

```
alpha3.B = post2.alpha.B
```

```
beta3.B = post2.beta.B
```

```
alpha3.S = post2.alpha.S
```

```
beta3.S = post2.beta.S
```

```
#posterior
```

```
post3.alpha.B = alpha3.B + d3.B
```

```
post3.beta.B = beta3.B + n3.B - d3.B
```

```
post3.alpha.S = alpha3.S + d3.S
```

```
post3.beta.S = beta3.S + n3.S - d3.S
```

```
#generating samples
```

```
set.seed(555)
```

```
postdist3.B = rbeta(n, post3.alpha.B , post3.beta.B)
```

```
postdist3.S = rbeta(n, post3.alpha.S , post3.beta.S)
```

```
p3.B.better = mean(postdist3.B < postdist3.S)
```

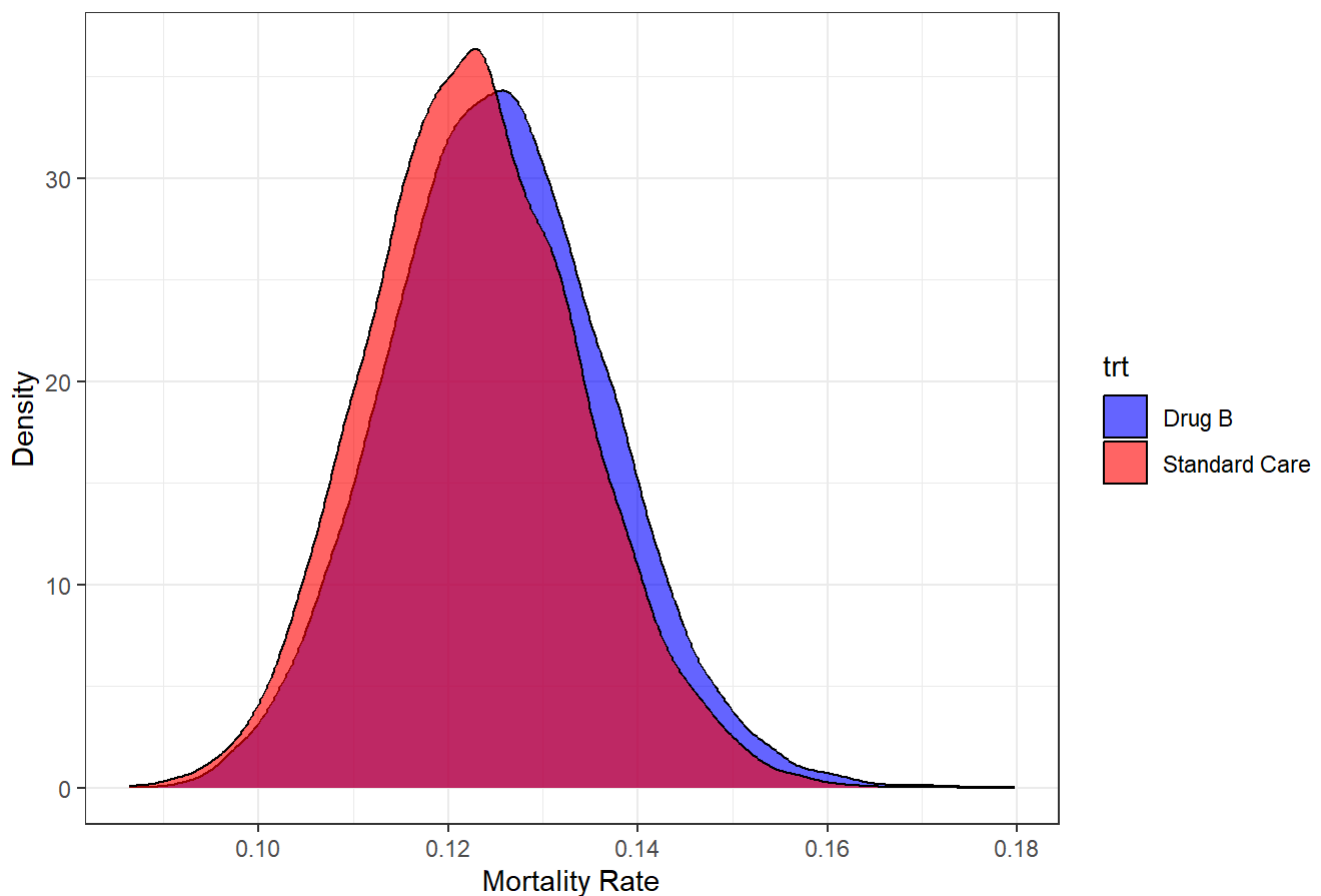
```
print(p3.B.better)
```

```
## [1] 0.4355
```

```
#plotting
data3.B.S = data.frame(
  d.r = c(postdist3.B , postdist3.S),
  trt = rep(c("B", "S"), times = c(n, n))
)

ggplot(data3.B.S, aes(x = d.r , fill = trt)) +
  geom_density(alpha = 0.6) +
  labs(title = "posterior distribution of death rate in treatment B ans SC",
       x = "Mortality Rate",
       y = "Density")+
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug B", "Standard Care"))+
  theme_bw()
```

posterior distribution of death rate in treatment B ans SC



```
#using MCMCpack
set.seed(555)
post3.B = MCbinomialbeta(d3.B, n3.B, post3.alpha.B, post3.beta.B, mcmc = n)
post3.S = MCbinomialbeta(d3.S, n3.S, post3.alpha.S, post3.beta.S, mcmc = n)
mean(post3.B < post3.S)
```

```
## [1] 0.456
```

```
#####
```

```
# drug C vs SC #
```

```
d3.C = 59
```

```
n3.C = 649
```

```
#prior
```

```
alpha3.C = post2.alpha.C
```

```
beta3.C = post2.beta.C
```

```
#posterior
```

```
post3.alpha.C = alpha3.C + d3.C
```

```
post3.beta.C = beta3.C + n3.C - d3.C
```

```
#generating samples
```

```
set.seed(555)
```

```
postdist3.C = rbeta(n, post3.alpha.C , post3.beta.C)
```

```
p3.C.better = mean(postdist3.C < postdist3.S)
```

```
print(p3.C.better)
```

```
## [1] 0.9949
```

```
#plotting
```

```
data3.C.S = data.frame(
```

```
  d.r = c(postdist3.C , postdist3.S),
```

```
  trt = rep(c("C", "S"), times = c(n, n))
```

```
)
```

```
ggplot(data3.C.S, aes(x = d.r , fill = trt)) +
```

```
  geom_density(alpha = 0.6) +
```

```
  labs(title = "posterior distribution of death rate in treatment C ans SC",
```

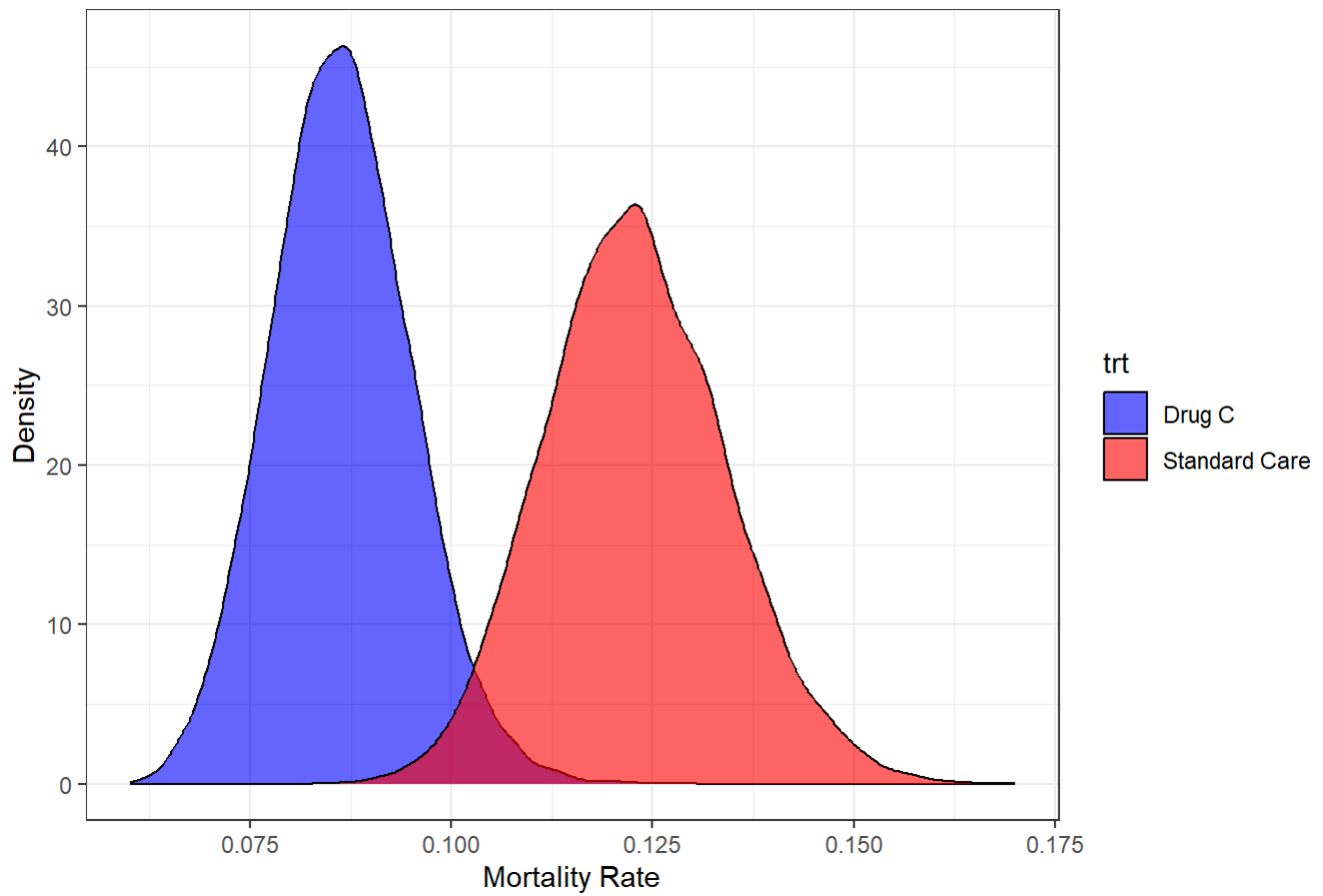
```
        x = "Mortality Rate",
```

```
        y = "Density")+
```

```
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug C", "Standard Care"))+
```

```
  theme_bw()
```

## posterior distribution of death rate in treatment C ans SC



```
#using MCMCpack
set.seed(555)
post3.C = MCbinomialbeta(d3.C, n3.C, post3.alpha.C, post3.beta.C, mcmc = n)
mean(post3.C < post3.S)
```

```
## [1] 1
```

```
#####
```

```
# drug D vs SC #
```

```
d3.D = 71
```

```
n3.D = 442
```

```
#prior
```

```
alpha3.D = post2.alpha.D
```

```
beta3.D = post2.beta.D
```

```
#posterior
```

```
post3.alpha.D = alpha3.D + d3.D
```

```
post3.beta.D = beta3.D + n3.D - d3.D
```

```
#generating samples
```

```
set.seed(555)
```

```
postdist3.D = rbeta(n, post3.alpha.D , post3.beta.D)
```

```
p3.D.better = mean(postdist3.D < postdist3.S)
```

```
print(p3.D.better)
```

```
## [1] 0.2151
```

```
#plotting
```

```
data3.D.S = data.frame(
```

```
  d.r = c(postdist3.D , postdist3.S),
```

```
  trt = rep(c("D", "S"), times = c(n, n))
```

```
)
```

```
ggplot(data3.D.S, aes(x = d.r , fill = trt)) +
```

```
  geom_density(alpha = 0.6) +
```

```
  labs(title = "posterior distribution of death rate in treatment D ans SC",
```

```
        x = "Mortality Rate",
```

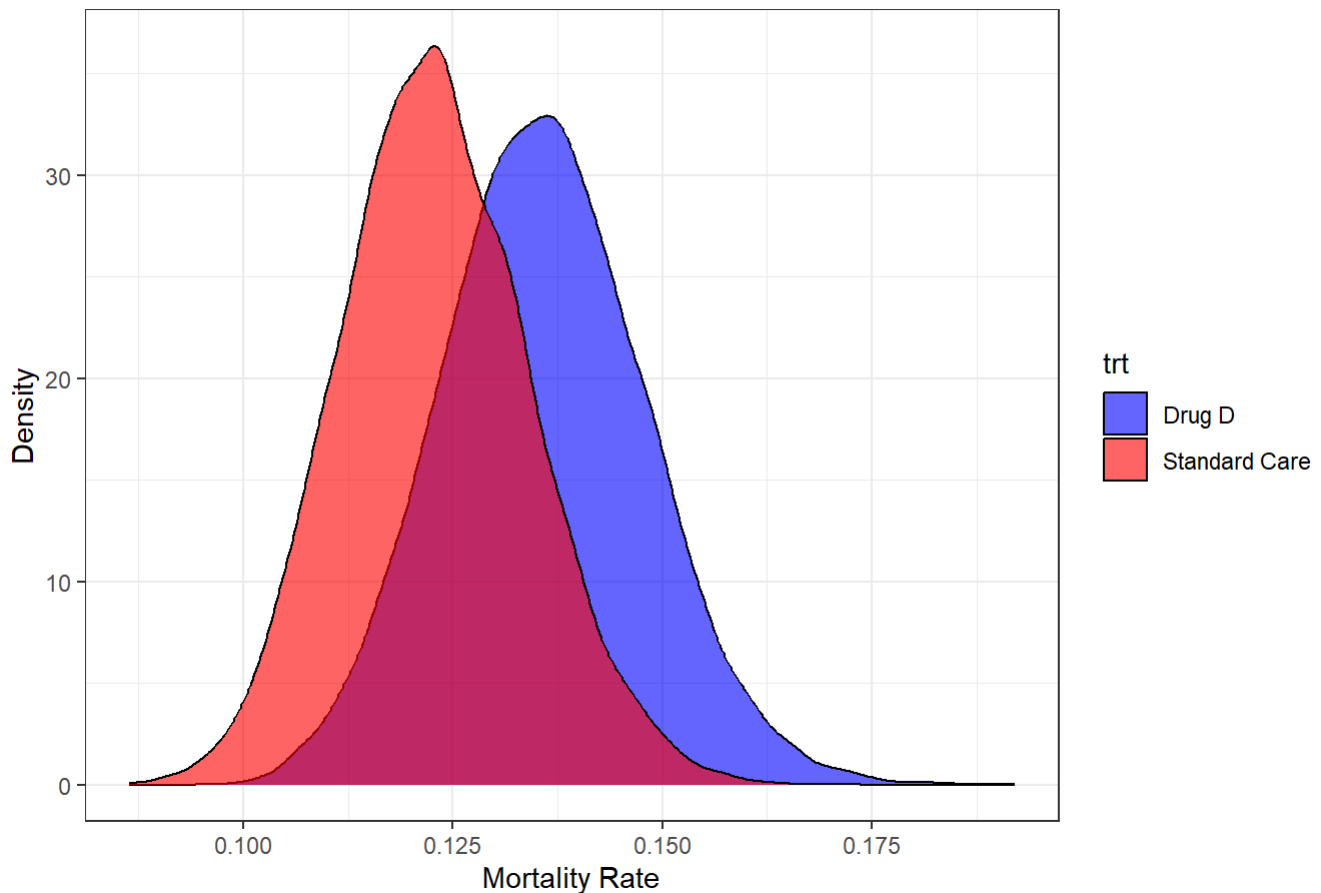
```
        y = "Density")+
```

```
  scale_fill_manual(values = c("blue", "red"), labels = c("Drug D", "Standard Care"))+
```

```
  theme_bw()
```



## posterior distribution of death rate in treatment D and SC



```
#using MCMCpack
set.seed(555)
post3.D = MCbinomialbeta(d3.D, n3.D, post3.alpha.D, post3.beta.D, mcmc = n)
mean(post3.D < post3.S)
```

```
## [1] 0.129
```

## Conclusion

Drug A, B, and D are concluded to be not better than SC and drug C is found to be better than SC in reducing mortality